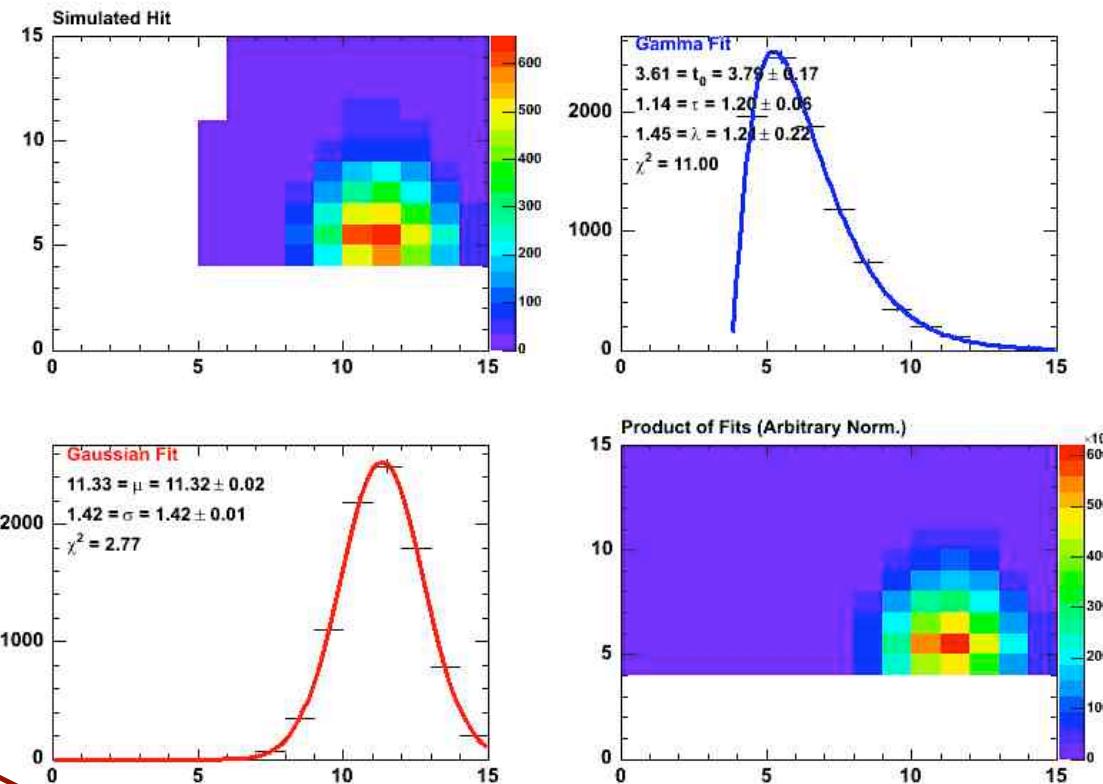


HitFinder Status - *tests on simulated hits*

- 1d fits in X & Y
 - Gaussian in X
 - $A \exp[-(x-\mu)^2/2\sigma^2]$
 - Gamma in Y (time)
 - $A y^\lambda e^{-y}; y=(t-t_0)/\tau$
- Same fn. used for simulation
 - random parameters
 - Poisson noise added
- Levenberg-Marquart
 - iterative interpolation
 - steepest decent ($\lambda \gg 1$)
 - quadratic approx ($\lambda \ll 1$)
- Starting values
 - A, μ, t_0, λ , from data
 - σ, τ , set to average values

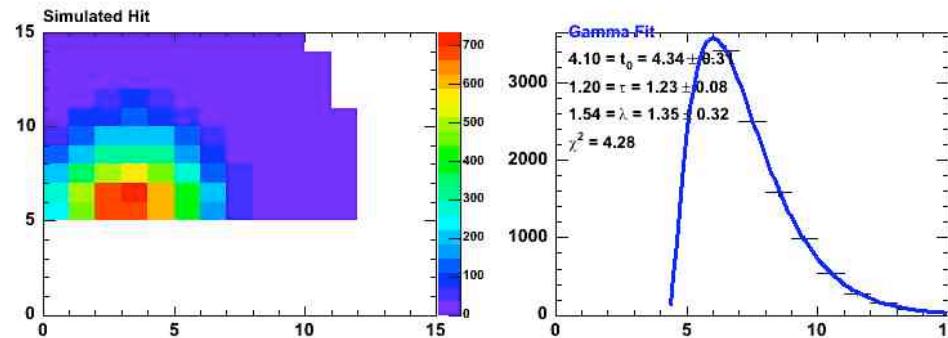


- Very Fast iteration (<100 lines of code)
- Used by E910 for y-pos. only
- Faster convergence for Gaussian
- *Requires good starting values*
 - *possible to try several sets*

Sample convergence - *fast*

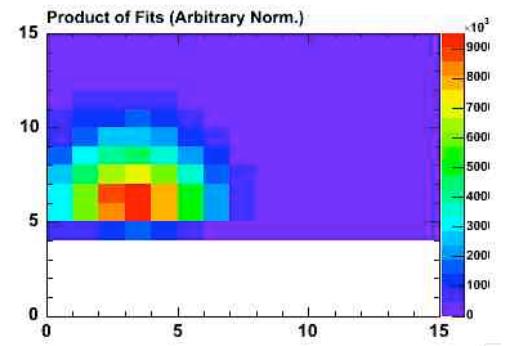
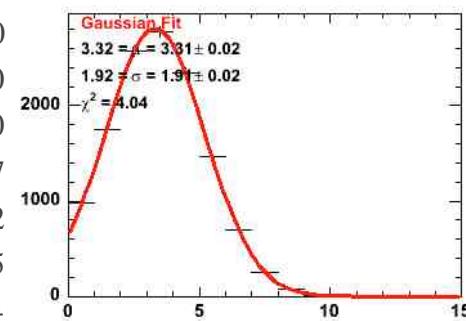
- Gaussian Fit

- 928.15 3.32 1.92 Input values
- 2771.84 3.50 1.50 Starting Values
- 2698.77 3.34 1.94 lam = 1e-04 chi2 = 2.0e+01
- 2804.20 3.31 1.91 lam = 1e-05 chi2 = 4.0e+00

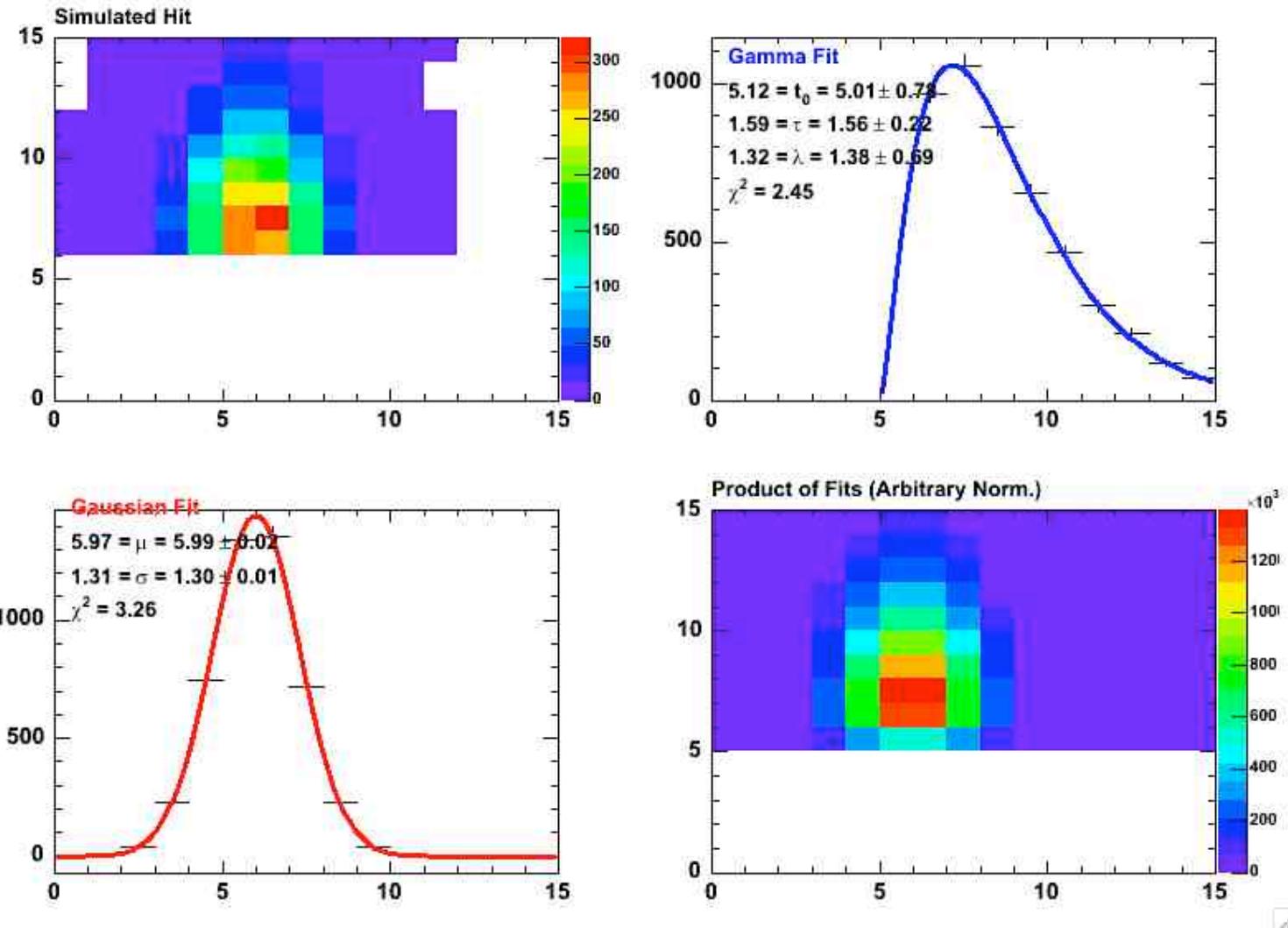


- Gamma Fit

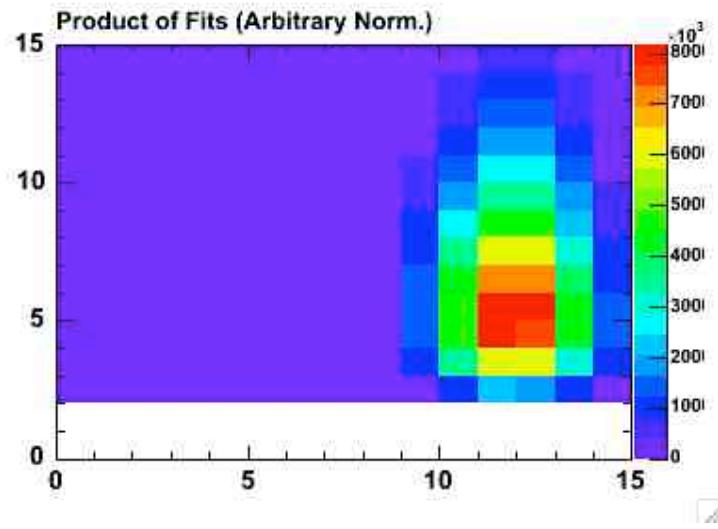
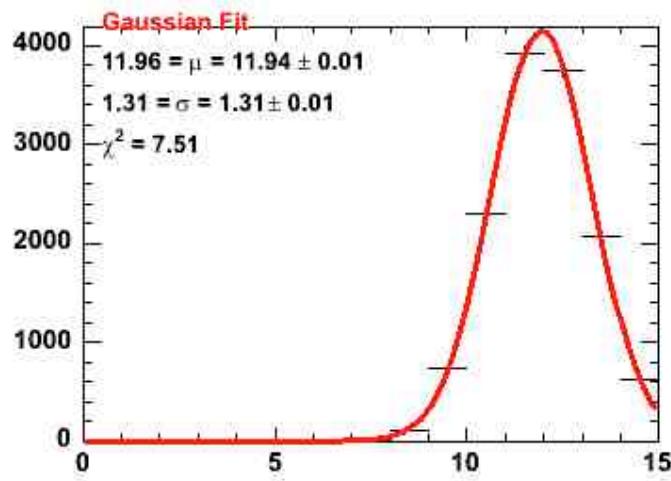
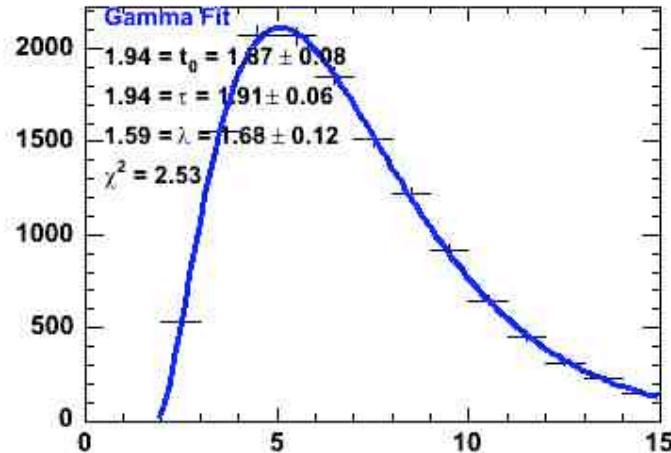
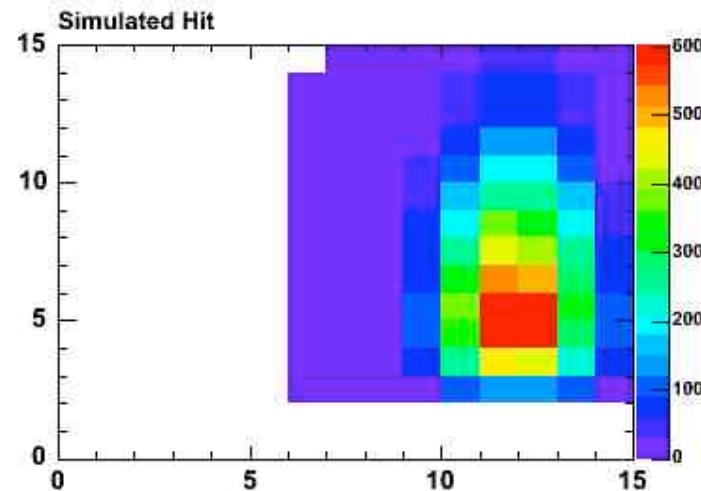
- Starting Values 9293.47 4.50 1.50 1.00
- lam = 1e-04 chi2 = 7.1e+00 9839.42 4.66 1.33 1.00
- lam = 1e-03 chi2 = 7.1e+00 9839.42 4.66 1.33 1.00
- lam = 1e-04 chi2 = 6.2e+00 9824.18 4.50 1.27 1.17
- lam = 1e-05 chi2 = 4.7e+00 9396.13 4.37 1.23 1.32
- lam = 1e-06 chi2 = 4.3e+00 9228.71 4.34 1.23 1.35
- Input values 4.10 1.20 1.54



Another sample fit



And another





Task List

- Test on real data
 - save real hit clusters to ascii format
 - fit with same program
- Insert in ana-mipp
 - Use of NR routines OK?
 - rewrite/debug takes time
- Tackle 2d
 - only when needed (1d is faster)
 - use tracking info to determine refit criteria (close tracks, near vertex)
 - generalize Levenberg-Marquart to 2d, or find other method